# uFR NDEF Console Application Example
# 1.0

# Table of contents

# Application preview



At the start of the application, user will be prompted with two options for opening communication with the connected reader.

Option 1 - Simple reader open, will be using ReaderOpen() function from our uFR API. This allows automatic searching and connecting to any device of uFR series that is currently connected via USB cable.

Option 2 - Advanced reader open, will prompt further options for opening communication. User will be prompted to enter following parameters:

Reader type

Port name

Port interface

Additional argument

Further explanation on these parameters can be found in uFR series API which you can find and download from the following link:

https://www.d-logic.net/code/nfc-rfid-reader-sdk/ufr-doc.git

# Main menu

Main menu of our application consists of following options:

```
-------------------------------------------
+---------------------------------------+
|            uFR NDEF example           |
+---------------------------------------+
-------------------------------------------
 (1) - Read NDEF records
 (2) - Write NDEF record
 (3) - Initialize card for NDEF messages
 (4) - Erase last NDEF record
 (5) - Erase all NDEF records
-------------------------------------------
press ESC and hit enter to exit.
```

On Windows, simply press 1 on your keyboard, or any other corresponding key, and application will proceed according to users selection.

As for Linux or MacOS platform, user will need to press 1 or any other corresponding key, and hit ENTER for the application to respond accordingly to users selection.

Pressing any other key than designated 1-5 keys, will prompt this Main menu screen with options.

# Reading NDEF records

This is selection 1 in our apps menu, simply choosing the option will read all NDEF records written on your NDEF card.

Successful reading should look like this:

```
-------------------------
Card type: DL_NTAG_213
Messages : 1
Records : 2
Empty records : 0
record_nr: 1
-------------------------
No: 1
Type: U
Length: 22
Payload: http://www.d-logic.net
record_nr: 2
-------------------------
No: 2
Type: U
Length: 17
Payload: tel:+381127195225
-------------------------
Reading done.
-------------------------------------------------
```

Here we can see information about our card type and number of messages and records on it.
And then we have them read in order one by one, displaying type of NDEF record, length of its data and its actual payload.

# Writing NDEF records

Selection 2 of our app. Here we have further option such as type of NDEF record we wish to write:

```
------------------------------------------------
Choose NDEF record type you wish to write:
    (1) - Phone
    (2) - SMS
    (3) - URI
    (4) - vCard
    (5) - Bluetooth
```

For each of these options user will be further prompted for input of additional parameters required for any of these NDEF types to be written.

For example for Phone NDEF we have a single input - phone number which we wish to write as NDEF record.

As for URI NDEF, for example, we have two inputs. First is a URI identifier:

```
Enter URI identifier you wish to write(''):
1 - http://www. | 2 - https://www. | 3 - http:// etc..
```

And second is URI link:

```
Enter URI link you wish to write (e.g d-logic.net | google.com | etc
d-logic.net
```

After our input, hit enter and our function should display information about our function being successful or not - status.

Status determines if our function has written data into card successfully or failed while writing due to some sort of error.

In case of a successful data write, our function will display: **"Status: UFR_OK"**

In any other case next to "Status: " will be displayed error message such as: **"Status: UFR_NO_CARD"** if our reader is not detecting any card in its field.

# Other options

Among other options, we have selections 3, 4 and 5. In other words, choices consisting of the following:

(3) – Card initialize for NDEF messages
(4) – Erase last NDEF record
(5) – Erase all NDEF records.

These functions are more or less self-explanatory and there is no input. Just select any of these options and following happens:

Selection 3:

Function prepares the card for NDEF using. Function writes Capability Container (CC) if necessary, and writes empty message. If card is MIFARE® CLASSIC or MIFARE® PLUS, then function writes MAD (MIFARE® Application Directory), and default keys and access bits for NDEF using.

Selection 4:

Self-explanatory: Deletes last written NDEF record from the card.

Selection 5:

Self-explanatory: Deletes all NDEF records from the card.

# R e v i s i o n   h i s t o r y

| Date | Version | Comment |
|---|---|---|
| 2019-04-09 | 1.0 | Base document |

Digital Logic Ltd.